

PATENT ABSTRACTS OF JAPAN

(11)Publication number : **08-272815**

(43)Date of publication of application : **18.10.1996**

---

(51)Int.CI.

**G06F 17/30**

**G06F 12/00**

---

(21)Application number : **07-077595**

(71)Applicant : **HITACHI LTD**

(22)Date of filing : **03.04.1995**

(72)Inventor : **YOKOYAMA TAKANORI**

**NAYA EIKO**

**SAITO MASAHIKO**

**HIROTA ATSUSHI**

**TSUNETOMI KUNIHIKO**

**KAMIWAKI TADASHI**

---

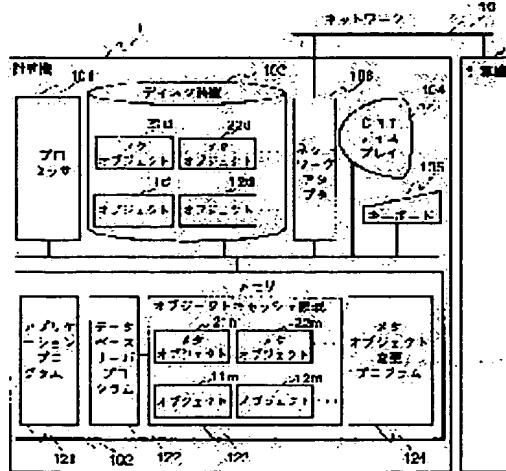
**(54) OBJECT-ORIENTED DATA BASE SYSTEM AND PROCESSING METHOD THEREFOR**

**(57)Abstract:**

**PURPOSE:** To easily change the operation of a system corresponding to the request of a user, a hardware and an input/output equipment, etc., by correcting only a meta object independently of the definition of an object.

**CONSTITUTION:** A meta object change program 124 is actuated and a programmer specifies the meta procedure of the meta object on a disk through a CRT 104 and a keyboard 105 and inputs the program of the meta procedure to be changed. For instance, the meta object 21m stores the leading address of the program of the meta procedure by a meta procedure address. The meta object change program 124 replaces the program of the execution form of the new meta procedure with the program of the specified leading address. Thereafter,

when a data base server program 122 is executed, the utilization of the changed meta procedure is made



**BEST AVAILABLE COPY**

possible.

(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平8-272815

(43)公開日 平成8年(1996)10月18日

(51)Int.Cl. <sup>6</sup> G 06 F 17/30 12/00	識別記号 5 4 7	府内整理番号 9194-5L 7623-5B	F I G 06 F 15/40 12/00	技術表示箇所 3 8 0 E 5 4 7 A
--	---------------	------------------------------	---------------------------------	------------------------------

審査請求 未請求 請求項の数16 O L (全20頁)

(21)出願番号 特願平7-77595	(71)出願人 株式会社日立製作所 東京都千代田区神田駿河台四丁目6番地 横山 孝典
(22)出願日 平成7年(1995)4月3日	(72)発明者 茨城県日立市大みか町七丁目1番1号 株式会社日立製作所日立研究所内 納谷 英光
	(72)発明者 茨城県日立市大みか町七丁目1番1号 株式会社日立製作所日立研究所内 斎藤 雅彦
	(74)代理人 弁理士 高橋 明夫 (外1名)
	最終頁に続く

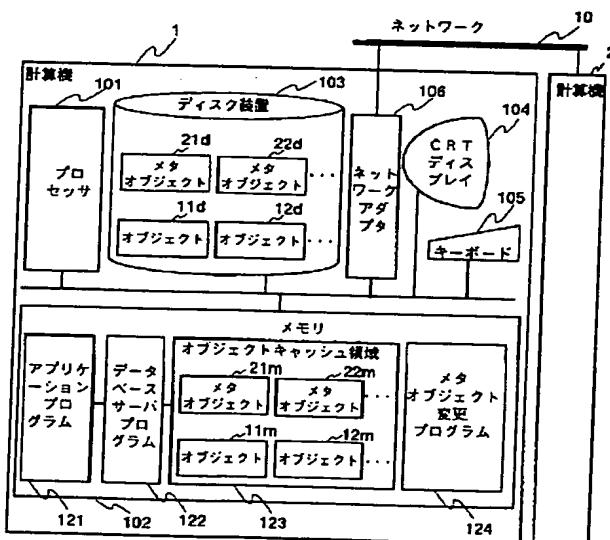
(54)【発明の名称】 オブジェクト指向データベースシステムおよびその処理方法

(57)【要約】

【目的】ユーザの要求やシステムの構造等に応じて、システムやオブジェクトの動作を変更できるオブジェクト指向データベースシステムを提供する。

【構成】メモリ102上に、データと手続きからなるオブジェクト11m等と、各オブジェクトに対応して、オブジェクトの構造や動作に関するメタデータと、オブジェクトに関する操作を行うメタ手続きを有するメタオブジェクト21m等を記憶する。本システムは、メタオブジェクト変更プログラム124によってメタ手続きを変更し、オブジェクトの生成時、データの書き込み前後、入出力手続き呼出し前後の各処理をカスタマイズ可能にする。例えば、データ書き込み後メタ手続きは、メモリ上のオブジェクトのデータの値や構造を変更した場合はそのつど、オブジェクトの手続きやメタオブジェクトを変更した場合はシステム立ち下げ時に、ディスク103に記憶し一致化する。また、本システムは、データ修正メタ手続きをユーザコマンドで呼び出して、オブジェクトデータの動的な修正(追加、削除等)を実行する。

図 1



## 【特許請求の範囲】

【請求項1】 データとそれに関する処理を記述した手続きを有するオブジェクトを、システム停止後も記憶可能な永続的記憶装置と一時的記憶装置の両方に記憶するオブジェクト指向データベースを備え、アプリケーションプログラムが前記オブジェクトにアクセスして処理を進めるオブジェクト指向データベースシステムにおいて、

各オブジェクトに対応して、オブジェクトの構造や動作に関する情報であるメタデータと、そのメタデータを参照してオブジェクトのデータ構造や手続きに関する処理を記述したメタ手続きを有するメタオブジェクトを、前記オブジェクト指向データベースに設けることを特徴とするオブジェクト指向データベースシステム。

【請求項2】 データ、それに関する処理を記述した手続き、その手続きの同じものをまとめた共有手続きを有する複数のオブジェクトを、システム停止後も記憶可能な永続的記憶装置と一時的記憶装置の両方に記憶するオブジェクト指向データベースを備え、アプリケーションプログラムが前記オブジェクトにアクセスして処理を進めるオブジェクト指向データベースシステムにおいて、複数のオブジェクトに対し、各オブジェクトの構造や動作に関する個別の情報である個別メタデータ、複数のオブジェクトに共通する情報である共有メタデータと、それらメタデータを参照してオブジェクトのデータ構造や手続きに関する処理を記述したメタ手続きを有するメタオブジェクトを、オブジェクト指向データベースに設けることを特徴とするオブジェクト指向データベースシステム。

【請求項3】 請求項1または2において、

前記メタオブジェクトは、前記一時的記憶装置と前記永続的記憶装置の間で、記憶するオブジェクトの内容を一致化する処理を記述したメタ手続きを有することを特徴とするオブジェクト指向データベースシステム。

【請求項4】 請求項1、2または3において、

前記一時的記憶装置または前記永続的記憶装置に記憶されている任意のメタ手続きを指定し、外部入力されるメタ手続きのプログラムをコンパイルし、実行形式に変換して、前記指定されたメタ手続きと入れ替えるメタオブジェクト変更手段を設けることを特徴とするオブジェクト指向データベースシステム。

【請求項5】 請求項1、2、3または4において、

前記メタデータは、前記オブジェクトのデータ及び手続きの記憶領域や前記手続きの呼び出し処理に関する情報を含み、前記メタ手続きは、前記メタデータを参照して前記オブジェクトの変更処理を記述したものを含み且つ、ユーザのコマンド入力により呼び出し可能に設けることを特徴とするオブジェクト指向データベースシステム。

【請求項6】 請求項5において、

前記オブジェクトの変更処理のメタ手続きは、オブジェクトの記憶領域の変更、オブジェクト内のデータの追加や削除、データ型の変更、手続きの追加や削除、手続きの修正またはレプリカの設置や移動を記述していることを特徴とするオブジェクト指向データベースシステム。

【請求項7】 データとそれに関する処理を記述した手続きを有するオブジェクトと、オブジェクトに対応してオブジェクトの構造や動作に関する情報であるメタデータ、そのメタデータを参照してオブジェクトのデータ構造や手続きに関する処理を記述したメタ手続きを有するメタオブジェクトを、システム停止後も記憶可能な永続的記憶装置と一時的記憶装置の両方に記憶し、アプリケーションプログラムが前記一時的記憶装置にアクセスして処理を進めるオブジェクト指向データベースシステムにおいて、

アプリケーションプログラムのアクセスにより実行中のオブジェクトから呼び出されるメタ手続きは、オブジェクトの生成時、アプリケーションプログラムのオブジェクトへのデータの書き込み前または書き込み後、オブジェクト内の手続き中での入出力手続き呼び出し前または呼び出し後の各タイミングに応じて、各自に記述されている異なる処理を実行することを特徴とするオブジェクト指向データベースシステムの処理方法。

【請求項8】 請求項7において、

前記データ書き込み前のメタ手続きは、所定のオブジェクトのデータにアクセスするために前記アプリケーションプログラムを起動したユーザのアクセス権を、メタデータを参照して判定し、アクセス否の場合は書き込み処理を行わないことを特徴とするオブジェクト指向データベースシステムの処理方法。

【請求項9】 請求項7または8において、

前記データ書き込み前のメタ手続きは、設定データの値が許容範囲内か、メタデータを参照して判定し、許容範囲外の場合は書き込み処理を行わないことを特徴とするオブジェクト指向データベースシステムの処理方法。

【請求項10】 請求項7において、

前記データ書き込み後のメタ手続きは、前記一時的記憶装置へのデータの書き替え直後に呼び出され、この書き替えられたオブジェクトのデータを前記永続的記憶装置の対応するデータ記憶領域に書き込むことを特徴とするオブジェクト指向データベースシステムの処理方法。

【請求項11】 請求項7において、

前記データ書き込み後のメタ手続きは、前記一時的記憶装置へのデータの書き替え直後に呼び出され、前記永続的記憶装置への前回の書き込みから一定時間経過したか判定し、経過している場合にその間に書き替えられたオブジェクトのデータを前記永続的記憶装置の対応するデータ記憶領域に書き込むことを特徴とするオブジェクト指向データベースシステムの処理方法。

【請求項12】 請求項10または11において、

前記データの書き換えに伴って変更されたメタデータは、システム立ち上げ時に前記永続的記憶装置の対応するメタデータ記憶領域に書き込むことを特徴とするオブジェクト指向データベースシステムの処理方法。

【請求項13】 請求項7において、

前記入出力手続き呼び出し後のメタ手続きは、実行された入出力手続きが正常に終了したか否かを、引数または戻り値により指定される情報を基に判定し、異常終了の場合は所定の異常対応処理を行うことを特徴とするオブジェクト指向データベースシステムの処理方法。

【請求項14】 データとそれに関する処理を記述した手続きを有するオブジェクトと、オブジェクトに対応してオブジェクトのデータや手続きの記憶領域等の構造に関する情報及び手続きの呼び出し動作に関する情報であるメタデータ、そのメタデータを参照してオブジェクトのデータ構造や手続きに関する処理を記述したメタ手続きを有するメタオブジェクトを、アプリケーションプログラムがアクセスして処理を進めるオブジェクト指向データベースシステムにおいて、データベースサーバープログラムを介したコマンド入力により呼び出されるメタ手続きは、オブジェクトの記憶領域の変更、オブジェクト内のデータの追加や削除、データ型の変更、手続きの変更（追加、削除または修正）、またはレプリカの設置や移動をコマンドの指定に応じて動的に実行することを特徴とするオブジェクト指向データベースシステムの処理方法。

【請求項15】 請求項14において、

前記オブジェクト内のデータの追加を行うメタ手続きは、追加する新しいデータの情報を渡されると、まず、対応するオブジェクトへのアクセスを禁止し、次に、新しいオブジェクトデータの記憶領域を割当て、メタデータをオブジェクトのデータに追加するように変更し、現在のオブジェクトデータを割当た前記記憶領域へコピーすることを特徴とするオブジェクト指向データベースシステムの処理方法。

【請求項16】 請求項14において、

前記オブジェクト内の手続きの変更を行うメタ手続きは、変更する新しい手続き情報を渡されると、まず、対応するオブジェクトへのアクセスを禁止し、次に、メタデータを参照して、対応するオブジェクトの手続きを前記新しい手続きとし、メタデータをオブジェクトの手続き変更に対応するように変更することを特徴とするオブジェクト指向データベースシステムの処理方法。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、オブジェクト指向データベースシステムに係り、特に、オブジェクトやシステムの動作、あるいは、オブジェクトを記憶するための記憶構造や記憶方法を、メタオブジェクトにより容易に修正できるオブジェクト指向データベースシステムとその

処理方法に関する。

【0002】

【従来の技術】 オブジェクト指向データベースでは、データとそれに関する手続きをまとめた、オブジェクトと呼ぶ単位でデータを記憶する。オブジェクト指向データベースでは、単なるデータのみを記憶するのではなく、データへのアクセスや、データ変換、入出力装置への入出力処理、探索処理等、そのデータに関する手続きも、データと一緒に記憶できる。また、オブジェクト間の関係も記憶できる。

【0003】 一般的のオブジェクト指向データベースでは、オブジェクトの生成方法や記憶方法は、各オブジェクト指向データベースのデータベース管理システムによって固定的である。そして、データベース管理システムの内部プログラムがオブジェクトの生成や、ディスク装置への書き込み、読み出しの処理を実行する。

【0004】 データベース管理システムの内部には、各オブジェクトの記憶位置や記憶構造、データのとりうる制約条件などを、メタデータと呼ばれる情報に記憶しており、内部プログラムはメタデータを参照しながらオブジェクトに関する処理を実行する。しかし、ユーザは、メタデータの詳細や、メタデータとオブジェクトの対応関係が分からず、個々のオブジェクトのデータの記憶位置や記憶構造、制約条件等を容易に知ることはできない。このため、メタデータを参照した処理を、ユーザが定義することはできなかった。

【0005】 また、データベース管理システムの内部プログラムやメタデータの項目やデータ構造は固定的であり、ユーザの要求やシステム構造に応じてそれを修正、変更するには、データベース管理システム全体を見直して改造する必要があり、多大な労力を要した。

【0006】 このようなオブジェクト指向データベースの問題点の解決案として、例えば、特開平6-83693に記載の方法がある。この第1の引用例では、オブジェクトのデータと一部のメタデータをまとめてオブジェクトとする。これにより、オブジェクトとメタデータの対応を容易にし、オブジェクトの手続きがメタデータの一部にもアクセス可能となり、一部のメタデータに関しては修正、変更が容易になる。

【0007】 また、「AI・IEE・IEE・IEE・Software」、第11巻、第3号（1994年）（" IEEE, SOFTWARE", Vol. 11, No. 3 (1994) ）、40頁～47頁に記載の方法がある。この第2の引用例では、オブジェクトの生成法や、同一種類（クラス）のオブジェクトに共通するデータ項目を定義したメタクラスを設け、そのメタクラスを、オブジェクトの定義と同一形式で定義可能としている。そして、メタクラスの定義を変更することにより、データベース管理システムの動作の一部を容易に修正可能としている。

【0008】

【発明が解決しようとする課題】しかし、第1の引用例の技術では、オブジェクトに一体化してアクセスできるメタデータは限定されており、オブジェクトの記憶位置や記憶構造などの情報は含まれない。したがって、オブジェクトの記憶方法を変更したり、ユーザの要求や使用ハードウェアに応じてカスタマイズすることは、容易にはできない。

【0009】また、本来、オブジェクトのデータには無関係なシステムを構成するハードウェアや使用する入出力機器に依存したメタデータもオブジェクト中に含まれてしまうため、異なるハードウェアや入出力機器を用いたシステムへ移植する場合に、オブジェクトの定義の修正が必要になるという問題がある。

【0010】また、上記第2の引用例の技術では、オブジェクトの生成処理以外のオブジェクトに対する操作については、その修正はできず、オブジェクトの記憶方法を変更することは容易ではない。また、生成処理についても修正を実行時に行えない。同一種類すなわち同一クラスに属する2つのオブジェクトの動作を異なるものにすることもできない。

【0011】このように、従来のオブジェクト指向データベースでは、システム変更の自由度はかなり限定されたものとなっている。このため、オブジェクトの記憶方法や、アクセス方法、エラー発生時の処理等を容易に修正することはできない。したがって、機能を限定して性能を向上させたいとか、性能よりも信頼性を重視したい等、ユーザの要求に応じた修正は困難であるという問題がある。また、システムのハードウェア構造、使用する入出力機器等に応じて、処理方法をカスタマイズし、性能や信頼性、メモリやディスクの使用効率等を向上させることも難しい。

【0012】本発明の目的は、ユーザの要求や、システムのハードウェア構造、使用する入出力機器等に応じて、オブジェクトを変更することなくメタオブジェクトによってカスタマイズできるオブジェクト指向データベースシステムを提供することにある。

【0013】本発明の目的は、オブジェクトの定義に影響を与えることなく、ユーザの要求や、システムのハードウェア構造、使用する入出力機器等に応じて、オブジェクトの記憶方法はじめ、システムや個々のオブジェクトの動作を容易に変更することができるオブジェクト指向データベースシステムの処理方法を提供することにある。

【0014】本発明のその他の目的は、以下の記載を通じて明らかになる。

【0015】

【課題を解決するための手段】本発明の目的は、データとそれに関する処理を記述した手続きを有するオブジェクトを、システム停止後も記憶可能な永続的記憶装置と一時的記憶装置の両方に記憶するオブジェクト指向データベースシステムである。

ターベスを備え、アプリケーションプログラムが前記オブジェクトにアクセスして処理を進めるオブジェクト指向データベースシステムにおいて、各オブジェクトに対応して、オブジェクトの構造や動作に関する情報であるメタデータと、そのメタデータを参照してオブジェクトのデータ構造や手続きに関する処理を記述したメタ手続きを有するメタオブジェクトを、前記オブジェクト指向データベースに設けることにより達成される。

【0016】また、上記構成において、前記一時的記憶装置または前記永続的記憶装置に記憶されている任意のメタ手続きを指定し、外部入力されるメタ手続きのプログラムをコンパイルし、実行形式に変換して、前記指定されたメタ手続きに入れ替えるメタオブジェクト変更手段を設けることを特徴とする。

【0017】本発明の他の目的は、アプリケーションプログラムのアクセスにより実行中のオブジェクトから呼び出されるメタ手続きは、オブジェクトの生成時、アプリケーションプログラムのオブジェクトへのデータの書き込み前または書き込み後、オブジェクト内の手続き中の入出力手続き呼び出し前または呼び出し後のタイミングに応じて、各自に記述されている異なる処理を実行することにより達成される。

【0018】また、データベースサーバープログラムを介したコマンド入力により呼び出されるメタ手続きは、オブジェクトの記憶領域の変更、オブジェクト内のデータの追加や削除、データ型の変更、手続きの変更（追加、削除または修正）、またはレプリカの設置や移動をコマンドの指定に応じて動的に実行することにより達成される。

【0019】

【作用】メタ手続きは、オブジェクトへのアクセス時や、オブジェクト内の手続きが入出力処理を行う時等、所定の時期に、暗黙的に起動される。また、システム管理者のコマンド入力により明示的に起動される。起動されたメタ手続きは、メタオブジェクト内に記憶されたメタデータを参照しながら、対応するオブジェクトのデータや手続きの記憶内容に対する操作や、オブジェクトの動作に伴ったエラー処理など、使用ハードウェアや入出力機器に依存した処理を実行する。

【0020】メタデータやメタ手続きはオブジェクトとは独立にメタオブジェクトで定義、記述される。したがって、オブジェクトの定義とは独立に、メタオブジェクトを修正するのみで、ユーザの要求や、ハードウェア、入出力機器等に応じて、容易にシステムの動作を変更することができる。

【0021】

【実施例】以下、本発明の実施例を図を用いて詳細に説明する。

【0022】図1は、本発明の第一の実施例によるオブジェクト指向データベースシステムの構成図である。複数のオブジェクト指向データベースシステムが接続され、各システムはオブジェクト指向データベースと接続している。

数の計算機、すなわち計算機1と計算機2がネットワーク10で結ばれている。計算機1はプロセッサ101、メモリ102、磁気ディスク装置103（以下単にディスクと呼ぶ）、CRTディスプレイ104、キーボード105、ネットワークアダプタ106から成る。

【0023】永続的な記憶装置であるディスク103には、オブジェクト11d、12d等、およびメタオブジェクト21d、22d等が永続的に記憶されている。各メタオブジェクトはそれぞれ一のオブジェクトに対応しており、メタオブジェクト21dはオブジェクト11dに、メタオブジェクト22dはオブジェクト12dに対応する。

【0024】メモリ102上には、アプリケーションプログラム121、データベースサーバのプログラム122、メタオブジェクト変更プログラム124が記憶されている。また、メモリ102上にはオブジェクトのキャッシュ領域123があり、ディスク103上のオブジェクトおよびメタオブジェクトから複製された複数のオブジェクト11m、12m等、メタオブジェクト21m、22m等が存在する。メモリは一時的な記憶装置であり、オブジェクトのキャッシュ領域123に記憶されたオブジェクトやメタオブジェクトの情報はシステム停止時に失われる。

【0025】プロセッサ101は、メモリ上のアプリケーションプログラム121、データベースサーバプログラム122、オブジェクト11m、12m等、メタオブジェクト21m、22m等のプログラムを読み出し、それらに含まれるデータにアクセスしながら処理を実行する。必要により、ユーザからのコマンドや処理に用いる情報がキーボード105から入力され、処理結果がCRTディスプレイ104に表示される。ネットワークアダプタ106は、ネットワーク10を介して他の計算機と情報をやりとりする。計算機2も計算機1と同様の構成である。

【0026】次にオブジェクト指向データベースシステムの動作の概要について説明する。データベースサーバプログラム122は、アプリケーションプログラム121がオブジェクトにアクセスする場合の仲介動作をする。また他の計算機上のオブジェクトにアクセスする場合、例えば計算機1上のアプリケーションプログラム121が計算機2上のオブジェクトにアクセスする場合は、データベースサーバプログラム122が、ネットワークアダプタ106を介し、ネットワーク10を経由して、計算機2上のデータベースサーバプログラムにアクセスを依頼する。そして依頼を受けた計算機2上のデータベースサーバプログラムが計算機2上のオブジェクトにアクセスする。これにより異なる計算機のオブジェクトにもアクセス可能となる。逆方向のアクセスも同様にして可能である。

【0027】ディスク103に記憶されているオブジ

クト11d、12d等、およびメタオブジェクト21d、22d等と、メモリ上のオブジェクトキャッシュ領域123に記憶されているオブジェクト11m、12m等、およびメタオブジェクト21m、22m等とは、それぞれ対応するもの同士の情報は同一である。本実施例のオブジェクト指向データベースシステムは立ち上げ時に、ディスク上に記憶されている全てのオブジェクトおよびメタオブジェクトの複製をオブジェクトキャッシュ領域に生成する。

【0028】アプリケーションプログラム121が、データベースサーバプログラム122を介して、オブジェクトキャッシュ領域123上の一つのオブジェクトにアクセスする場合、そのオブジェクトに含まれる手続きが実行される。

【0029】本実施例では、後述するように、オブジェクトキャッシュ領域123上のオブジェクトのデータの値が書き替えられたときやデータ構造が変更された時は、メタオブジェクトにより、その都度、ディスク103上の対応するオブジェクトの記憶情報も書き替える。

20 これにより、オブジェクトキャッシュ領域123上のオブジェクトの記憶情報と、ディスク103上のオブジェクトの記憶情報を常に一致させ、オブジェクトのデータ喪失を防止する。

【0030】しかし、オブジェクトの手続きが変更された場合には、ディスク103上の対応するオブジェクトの手続き情報は書き替えない。また、オブジェクトキャッシュ領域123上のメタオブジェクトの情報が書き替えられたときも、ディスク103上の対応するメタオブジェクトの記憶情報は書き替えない。そして、システム立ち上げ時に、オブジェクトキャッシュ領域123上の全てのオブジェクトの情報およびメタオブジェクトの情報が、ディスク103上に書き込まれる。

【0031】次にオブジェクトについて説明する。図2に、オブジェクトの構造を示す。上述のように、メモリ102上のオブジェクトとディスク103上のオブジェクトの記憶内容は基本的に同一であるので、以下では、メモリ上のオブジェクトについて説明する。

【0032】オブジェクトはデータとそれに関する手続きの集合である。図2では、オブジェクト11mは、データ11d1、11d2等と、手続きアドレス11p1、11p2等から成る。手続きアドレスは、手続きのプログラムの先頭アドレスを指定したものである。同図では、手続きアドレス11p1は手続きのプログラム11p10の先頭アドレス、手続きアドレス11p2は手続きのプログラム11p20の先頭アドレスを示している。例えば、オブジェクト指向言語C++でオブジェクトを定義すると、データは構造体、手続きはメンバ関数と呼ばれる関数で表現できる。

【0033】オブジェクト指向データベースシステムでは、外部のプログラム（アプリケーションプログラム）

がオブジェクト内のデータを直接読み出したり書き込んだりすることはできない。すなわち、外部のプログラムは、アクセスしようとするオブジェクトに含まれる手続きを呼びだし、その手続きが当該オブジェクト内のデータにアクセスする。

【0034】本実施例の場合、アプリケーションプログラム121が、データベースサーバプログラム122を介して、オブジェクトに含まれる手続き1、手続き2等を呼びだすことになる。手続きの呼び出しは、オブジェクトのデータに記憶されている手続きアドレス11p1、11p2等のアドレスを指定して実行する。すると、データベースサーバは、手続きアドレスのアドレスで指定された領域に記憶された手続きアドレスを指定して呼び出しを行う。これにより手続きのプログラムが実行される。

【0035】次に、メタオブジェクトについて説明する。図3に、メタオブジェクトの構造を示す。本例では、一つのオブジェクト11mに対し一つのメタオブジェクト21mが対応している。オブジェクトが有するデータの一つはメタオブジェクトへのアドレスを表わしており、これによりオブジェクトに対応するメタオブジェクトを知ることができる。メタオブジェクトはオブジェクトの生成時に同時に生成し、オブジェクト削除時に同時に削除する。

【0036】メタオブジェクトは、対応するオブジェクトのデータ構造や手続き、状態等に関する情報であるメタデータと、オブジェクトの操作やそれに関連する処理を行うメタ手続きの集合である。

【0037】図3のように、メタオブジェクト21mは、メタデータ21d1、21d2等と、メタ手続きアドレス21p1、21p2等から成る。メタ手続きアドレスは、メタ手続きのプログラムの先頭アドレスを指定したものである。メタ手続きアドレス21p1はメタ手続きのプログラム21p10の先頭アドレスを、メタ手続きアドレス21p2はメタ手続きのプログラム21p20の先頭アドレスを示している。

【0038】メタオブジェクトはオブジェクトと同様に、例えばオブジェクト指向言語C++でメタオブジェクトを表現すると、メタデータは構造体（個々のメタデータ又はメンバ）、メタ手続きはメンバ関数で表現できる。

【0039】次に、メタデータの詳細について説明する。図4、図5は、メタデータの具体的な内容及びデータ構造を示したものである。

【0040】メタデータは、対応するオブジェクトをシステム内で一意に識別するためのオブジェクト識別子21d1、当該オブジェクトが記憶されているメモリ領域の先頭アドレス21d2、オブジェクトを記憶するメモリ領域のサイズ（バイト数）21d3、当該オブジェクトにアクセスできるユーザの数21d4、アクセス可能

なユーザに関する情報を記憶したアクセス可能者テーブル21d50へのポインタ（アドレス）21d5、オブジェクトのバージョン番号21d6、オブジェクトが有するデータの数21d7、そのデータに関する情報を記憶したデータ情報テーブル21d80へのポインタ21d8、オブジェクトが有する手続きの数21d9、その手続きに関する情報を記憶した手続き情報テーブル21d100へのポインタ21d10、オブジェクトが破壊されたときのバックアップに用いられるオブジェクトのレプリカ数21d11、レプリカの存在位置に関する情報を記憶したレプリカ位置情報テーブル21d120へのポインタ21d12、オブジェクトを作成したユーザ名21d13、当該オブジェクトを参照しているオブジェクトの数（参照数）21d14、参照しているオブジェクトの情報を記憶した参照オブジェクト情報テーブル21d150へのポインタ（アドレス）21d15、オブジェクトのディスク上の格納位置情報21d16等の情報を含んでいる。

【0041】アクセス可能者テーブル21d50は、オブジェクトにアクセス可能なユーザ名21d51、21d52等から成る。メタオブジェクトはこの情報を利用して、当該オブジェクトに対してアクセス権のないアプリケーションプログラムによるアクセスを阻止することができる。

【0042】データ情報テーブル21d80は、データ情報21d81、21d82等から成る。データ情報21d81は、データ名21d811、データ型名21d812、データの記憶領域のサイズ（バイト数）21d813、データの記憶領域の先頭アドレス21d814、データの値のとりうる最小値21d815、データの値のとりうる最大値21d816、データがこれまで何回書き替えられたかを表す書き込み回数21d817等から成る。メタオブジェクトは、これらのデータ情報をを利用して、オブジェクトのデータの内容や、データの記憶領域や記憶方法を制御することができる。また、データのアクセス回数に応じた処理も実現できる。

【0043】レプリカ位置情報テーブル21d120は、レプリカが存在する計算機の計算機アドレス21d121、21d122等から成る。参照オブジェクト情報テーブル21d150は、このオブジェクトを参照しているオブジェクトのアドレス21d151、21d152等から成る。メタオブジェクトは、これらの情報をを利用して、オブジェクトのレプリカを生成するとともに、レプリカの記憶位置や記憶方法を制御することができる。

【0044】手続き情報テーブル21d100は、手続き名毎に構成される手続き情報21d101、21d102等から成る。図5に、手続き情報の詳細な構造を示す。

【0045】手続き情報21d101は、手続き名21

d1011、オブジェクト内の手続きアドレスを記憶している領域のアドレス21d1012、手続きの記憶領域のサイズ(バイト数)21d1013、戻り値のデータ型名21d1014、戻り値のサイズ(バイト数)21d1015、引数の数21d1016、引数情報テーブル21d1070へのポインタ21d1017、当該手続き内で実行する入出力処理に使用する入出力手続き数21d1018、入出力手続き情報テーブル21d10190へのポインタ21d1019等から成る。

【0046】引数情報テーブル21d10170は、引数情報21d10171、21d10172等から成る。引数情報21d10171は、引数名21d101711、引数のデータ型名21d101712、引数のサイズ(バイト数)21d101713等から成る。

【0047】入出力手続き情報テーブル21d10190は、入出力手続き情報21d10191、21d10192等から成る。入出力手続き情報21d10191は、入出力手続き名21d101911、入出力手続きのアドレス21d101912、入出力手続きでエラーが生じた場合に実行するエラー処理のアドレス21d101913等から成る。

【0048】メタオブジェクトは、これらの手続き情報を利用して、オブジェクトの手続きのプログラムや、手続きの実行方法を制御することができる。また、入出力手続きに関するエラー処理等の付加的な処理を実行することができる。

【0049】このようなメタデータは、メタオブジェクトの定義により与えられる情報、オブジェクトのコンパイル時に抽出できる情報、オブジェクト生成時に決定される情報、オブジェクトの実行によって決まる情報に分類できる。

【0050】メタオブジェクトの定義により与えられる情報には、アクセス可能者情報21d5や、データのとりうる最小値21d815や最大値21d816がある。例えばメタオブジェクトの定義において、アクセス可能なユーザを指定することにより、アクセス可能者情報を定義できる。

【0051】オブジェクトのコンパイル時に抽出できる情報には、オブジェクトのサイズ21d3やデータ数21d7、データ名21d811、データ型21d812、手続き数21d9、手続き名21d1011、手続きのサイズ21d1014等がある。コンパイル時の情報の抽出については、先に本願発明者等が提案し、特開平5-313919号公報に開示されている方法に詳しい。

【0052】オブジェクト生成時に決定される情報には、オブジェクト識別子21d1やオブジェクトのアドレス21d2、データのアドレス21d814等がある。これらは後述するオブジェクト生成時に呼び出されるメタ手続きにより設定する。

【0053】オブジェクトの実行時に決まる情報には、参照オブジェクト情報21d15やデータの書き込み回数21d817等がある。これらはオブジェクトの実行時に呼び出されるメタ手続きにより設定する。例えばデータの書き込み終了後に呼び出されるメタ手続きにより、データの書き込み回数を更新する。また、ユーザがデータベースサーバにコマンドを入力して、メタ手続きを呼び出し、そのメタ手続きの処理内でメタデータを設定したり、変更したりすることもある。これについて後述する。

【0054】次に、メタ手続きの詳細について説明する。メタ手続きはオブジェクトの実行に伴って、所定のタイミングで呼び出されたり、データベースサーバプログラムへのユーザのコマンド入力により呼び出される。そして、メタデータにアクセスしながら処理を実行する。

【0055】オブジェクトの実行に伴って呼び出されるメタ手続きは、オブジェクトの生成時すなわちメタオブジェクトの生成時、アプリケーションプログラムの(データベースサーバプログラムを介した)オブジェクトへのデータの書き込み前、書き込み後、オブジェクト内の手続き中での入出力手続き呼び出し前、呼び出し後等に呼び出される。

【0056】オブジェクトの生成時に呼び出されるメタ手続きは、メタオブジェクトのメタデータの初期化を行う。すなわち、前述のメタデータのうち、オブジェクト識別子21d1やオブジェクトのアドレス21d2、データのアドレス21d814等、オブジェクト生成時に決まるデータを設定する。

【0057】オブジェクトのデータ書き込み前に実行されるメタ手続きは、アクセス権のチェックや、設定するデータが許される範囲内であるかどうかなどのチェックを行う。

【0058】オブジェクトのデータ書き込み後に実行されるメタ手続きは、そのオブジェクトに対応するディスク103上のオブジェクトにもデータを書き込む一致化処理を行う。これにより、オブジェクトキャッシュ領域123上のオブジェクトの情報と、ディスク103上のオブジェクトの記憶内容の一致化が実現できる。

【0059】入出力手続き呼び出し前に呼び出されるメタ手続きは、入出力データのチェック等を行う。入出力手続き呼び出し後に呼び出されるメタ手続きは、入出力処理のエラー処理等を行う。

【0060】本実施例では、メタオブジェクト変更プログラムを有していて、メタ手続きをプログラマが変更することができる。メタ手続きの変更には、アプリケーションプログラム121やデータベースサーバープログラム122の実行を停止する必要がある。この停止により、メモリ102上のオブジェクトやメタオブジェクトの状態はディスク103に反映される。

【0061】次に、メタオブジェクト変更プログラム124を起動し、プログラマがCRT104やキーボード105を介して、ディスク上のメタオブジェクトのメタ手続きを指定し、変更するメタ手続きのプログラムを入力する。たとえば、メタオブジェクト21mは、メタ手続きアドレス21p1により、メタ手続きのプログラム21p10の先頭アドレスを記憶している。メタオブジェクト変更プログラム124は、新たなメタ手続きの実行形式のプログラムを、指定された先頭アドレスのプログラム21p10に入れ替える。この後、データベースサーバプログラム122を実行すると、変更したメタ手続きの利用が可能になる。

【0062】このようなメタ手続きの変更により、オブジェクト生成時の処理、オブジェクトへのデータの書き込み前、後の処理、あるいはオブジェクト内の手続き中の入出力手続き呼出し前、後の処理を容易にカスタマイズできる。

【0063】また、異なるメタ手続きを有するオブジェクトを、システム内に同時に存在させることができる。これにより、オブジェクトに応じて、オブジェクトの生成時の処理、オブジェクトへのデータ書き込み前、後の処理、オブジェクト内の手続き中の入出力手続き呼出し前、後の処理等を任意に変えることができる。

【0064】以上、本実施例のメタ手続きについて説明した。メタ手続きは、メタデータを参照しながら、メモリ上のオブジェクトのデータとディスク上のオブジェクトのデータの一致化処理や、オブジェクトのデータや手続きの変更などオブジェクトに対する操作や、オブジェクトの処理に伴う計算機構成や入出力機器に依存したエラー処理などの付加的な処理を実行する。

【0065】次に、これらメタ手続きによる動作のいくつかを具体的に説明する。

【0066】図6は、データ書き込み前メタ手続きの動作を示すフローチャートである。データ書き込み前メタ手続きは、書き込みの直前に引数として設定するデータのアドレスと値が渡される。

【0067】データ書き込み前メタ手続き21p001は、まず、データにアクセスするアプリケーションプログラムを起動したユーザが、アクセス可能であるかどうかを判定する(処理21p0011)。これは、当該ユーザが、アクセス可能者テーブル21d50に記憶されているかどうかで判定する。アクセス不可であれば書き込みエラーとして終了する。

【0068】アクセス可能であれば、次に設定値が最小値以上であるかどうかを判定する(処理21p0012)。これには、データ情報(21d81、21d82等)をサーチし、そこに記憶されているデータのアドレスから引数で渡されたデータのアドレスと一致するもの(例えば21d814)を選択し、その最小値情報(例えば21d816)を参照し、これと設定する値と比較

する。最小値以下であれば書き込みエラーとして終了する。最小値以上であれば、次に設定値が最大値以下であるかどうかを判定する(処理21p0013)。これも同様にして、設定する値と、メタデータのうちの最大値情報(例えば21d817)を参照し、比較する。最大値以上であれば書き込みエラーとして終了する。最大値以下であれば正常終了する。

【0069】正常終了した場合には、オブジェクトの手続きに戻り、実際にデータの書き込みが実行される。一方、書き込みエラーとして終了すると、オブジェクトの手続きを呼び出したデータベースサーバプログラム122の呼び出しがエラーとして処理される。このように、メタオブジェクトでアクセス権やデータ値の制約条件をチェックするので、オブジェクトの定義を修正することなく、メタオブジェクトの関連するメタデータを書き替えるのみで、容易にアクセス権や制約条件を変更できる。

【0070】図7は、データ書き込み後メタ手続きの動作を示すフローチャートである。データ書き込み後メタ手続きは、データ書き込みの直後に、書き替えられたデータをディスク上の対応するオブジェクトのデータの記憶領域に書き込む。

【0071】データ書き込み後メタ手続き21p101は、オブジェクトのディスク103上の格納位置21d16と、データ情報12d81中のデータのアドレス21d84を参照し、書き込みデータでのディスク上への格納位置を決定する(処理21p1011)。そして、決定した格納位置に書き込みデータを格納する(処理21p1012)。

【0072】これにより、メモリ上のデータとディスク上のデータの一貫化処理が実行でき、システムが故障等により突然停止しても、最新のオブジェクトのデータが保存され、データが失われることがない。

【0073】ところで、オブジェクトデータが書き替えられる度に、一致化処理していたのでは処理性能上の問題がある。そこで、図8に示すように、前回書き込み後一定時間経過した場合に書き替えるようとしてもよい。

【0074】第2のデータ書き込み後メタ手続き21p102は、まず、前回のディスクへの書き込み後、一定時間経過したかどうかを調べる(処理21p1021)。経過している場合は、図7と同様の処理により書き込みデータを格納する。経過していない場合には、格納処理は実行しない。

【0075】次に、入出力後メタ手続きの動作を、図9のフローチャートにより説明する。入出力手続きの終了の直後に、入出力後メタ手続き21p201が呼び出され、引数として実行した入出力手続きのアドレスと、実行した入出力手続きが正常終了したかどうかの情報が渡される。

【0076】入出力後メタ手続き21p201は、ま

ず、実行した入出力手続きが正常終了したかどうかを引数により判定する（処理21p2011）。正常終了であれば、そのまま終了する。正常でなければ、入出力処理判定（処理21p2012）を実行する。すなわち、メタデータ内の入出力手続き情報（21d10191、21d10192等）をサーチし、そこに記憶されている入出力手続きのアドレス（例えば21d101912）を参照し、これと引数で渡された実行入出力手続きのアドレスを比較し、一致するものを持つ入出力情報を取得する。一致する入出力情報（例えば21d1091）を得られたら、それに含まれるエラー処理のアドレス（例えば21d101913）を得、そのエラー処理を呼び出す（処理21p2013）。

【0077】これにより、オブジェクトにエラー処理を記述することなく、メタオブジェクトで実行できる。また、オブジェクトを修正することなく、メタオブジェクトのエラー処理のアドレスを書き替えるのみで、使用する入出力機器に応じたエラー処理に変更することもできる。

【0078】この入出力後メタ手続きについても、プログラマがその動作を変更することができる。図10に、第2の入出力後メタ手続きの例を示す。入出力後メタ手続き21p202は、実行した入出力手続きが正常終了したかどうか引数により判定し（処理21p2021）、正常終了であれば本処理を終了してオブジェクトの処理に戻る。正常終了でなければ、システムが故障したものとして、システム停止処理を実行し（処理21p2022）、オブジェクトの処理に戻らずにシステムを停止する。

【0079】このように、本実施例のメタ手続きの変更により、オブジェクトの定義に影響を与えることなく、オブジェクトに設定するデータのチェックや入出力処理のエラー処理等を追加でき、ユーザの要求や、システムのハードウェア構造、使用する入出力機器等に応じて、システム動作の柔軟な変更が可能になる。

【0080】本実施例では、ユーザがコマンド入力により、データベースサーバプログラムから呼び出して実行するメタ手続きがあり、これによって、オブジェクトの構造や動作を変更することができる。このメタ手続きに、オブジェクトの記憶領域の変更や、オブジェクト内のデータの追加や削除、データ型の変更、手続きの追加や削除、手続きの修正、レプリカの設置や移動等を行うメタ手続き等がある。

【0081】従って、これらのメタ手続きを起動するコマンドをユーザに提供することにより、システムを停止することなく、システムの動作や構成を変更することができる。また、メタ手続きを変更することにより、上記のオブジェクトの変更方法そのものを、目的に合わせて変更することができる。以下、オブジェクトの構造や動作を変更するメタ手続きの例をいくつか説明する。

【0082】図11は、オブジェクト内のデータを追加するデータ追加メタ手続きの動作を説明するフローチャートである。

【0083】データ追加メタ手続き21p301は、引数として追加後のオブジェクトのサイズ（バイト数）および新しいデータのデータ情報が渡される。引数で渡されるデータ情報は、メタデータに含まれるデータ情報21d81と同じ形式をしている。データ情報は、メタデータの抽出と同様に、新しいデータを定義したオブジェクトのコンパイル時に抽出できる。

【0084】まず、対応するオブジェクトへのアクセスを禁止する（処理21p3011）。これはオペレーティングシステムが提供する排他制御機能を利用して実現できる。次に、新しいオブジェクトを記憶する領域を割り当てる（処理21p3012）。これはOSが提供するメモリ領域割り当て機能を利用して実現できる。次に、メタデータを変更する（処理21p3013）。すなわち、メタデータ内のデータ数を一つ増やすとともに、引数で与えられたデータ情報へのポインタを、データ情報テーブルの最後の要素の後ろに追加する。ただしデータのアドレスは引数で与えられたデータ情報では未定義なので、データ情報を参照して、新しく追加したデータの記憶位置を算出して格納する。

【0085】そして、現在オブジェクトデータを記憶している領域から、新たに割り当てる領域へ、オブジェクト内の情報をコピーする（処理21p3014）。このとき、新たにデータが追加されたので、手続きアドレスをコピーする領域は、追加したデータの記憶に必要な領域の後ろにする。そして、当該オブジェクトを参照している全てのオブジェクトについて、記憶している参照アドレスを、新たに割り当てるアドレスに書き替える（処理21p3015）。この処理は、メタデータ中の参照オブジェクト情報21d15の参照しているオブジェクトのアドレスを読み出して、参照しているオブジェクトのメタオブジェクトに対し、参照アドレスの変更を要求することにより実現する。これにより、オブジェクトの参照関係が正しく保たれる。また、このときメタデータ内のオブジェクトのアドレスも書き替える。そして、新しいオブジェクトを、ディスク装置103上に書き込み、ディスク上のオブジェクトの情報との一致化を行う（処理21p3016）。最後に、オブジェクトへのアクセス禁止を解除する（処理21p3017）。これにより、データ追加前のオブジェクトが、新たに追加されたデータの記憶領域を含むオブジェクトに置き換えられる。

【0086】このようなデータ追加メタ手続きにより、オブジェクトに動的にデータを追加することが可能になる。また、メモリ上のオブジェクトのデータ構造が変化した場合には、その都度最新のオブジェクトのデータ構造がディスク上保存される。

【0087】一方、この変更により生じたメタデータの変更是、システム立ち下げ時に反映するようにしている。これは、オブジェクトのデータ構造に関するメタデータは、オブジェクトのコンパイル時に抽出でき、システム停止によって失われることがないからである。ただし、次回のシステムの立ち上げ時に、メタデータを修正し、再生する処理が必要になる。

【0088】この再生処理を不要にするため、図12に示すように、オブジェクトのデータ構造を変更する度に、メタデータの変更をディスクに反映させるようにしてもよい。第2のデータ追加メタ手続き21p302は、基本的には図11と同様の処理となるが、メタデータの変更(21p3023)後に、ディスク上へのメタデータ格納(21p3024)が行われる。

【0089】図13は、手続き変更メタ手続きの動作を説明するフローチャートである。手続き変更メタ手続き21p401は、引数として新しい手続きの手続き情報を渡され、オブジェクト内の手続きの変更を行う。引数で渡される手続き情報は、メタデータに含まれる手続き情報21d101と同形式をしている。手続き情報は、メタデータの抽出と同様に、新しいメタ手続きのコンパイル時に抽出できる。

【0090】手続き変更メタ手続き21p401は、まず、対応するオブジェクトへのアクセスを禁止する(処理21p4011)。次に、メタデータ内の手続き情報(21d101、21d102等)を調べ、引数で与えられた手続き情報に含まれる手続き名と同一の手続き名で、メタデータ内の手続き情報を記憶している手続きアドレスのアドレスを得る(処理21p4012)。

【0091】そして、該手続きアドレスのアドレスが示しているオブジェクト内の手続きアドレス(例えば11p1、11p2等)の内容を、引数で与えられた手続きのプログラムの先頭アドレスに置き換える(処理21p4013)。そして、メタデータ内の手続き情報を引数で与えられた手続き情報に置き換える(処理21p4014)。最後に、オブジェクトへのアクセス禁止を解除する(処理21p4015)。

【0092】以上のように、変更対象オブジェクトの、変更対象手続きに対応する手続きアドレスが指している手続きのプログラムの先頭アドレスが、新しい手続きのプログラムの先頭アドレスに書き替えられ、以後、オブジェクトの該手続きの呼び出しがなされると、新しい手続きが実行されることになる。このような手続き変更メタ手続きにより、オブジェクトの手続きを、動的に置き換えることが可能となる。なお、上記した各処理で、引数により渡される情報は、戻り値によってもよい。

【0093】以上、本実施例のメタ手続きの動作について説明した。メタ手続きは、メタデータを参照しながら、メモリ上のオブジェクトのデータとディスク上のオブジェクトのデータの一致化処理や、オブジェクトのデ

ータや手続きの変更などオブジェクトに対する操作や、オブジェクトの処理に伴う計算機構成や入出力機器に依存したエラー処理などの付加的な処理等を実行することができる。そして、メタ手続きを変更することにより、それらの処理をプログラマが自由にカスタマイズできる。

【0094】本発明の一実施例の構成によれば、オブジェクトが個別にデータおよび手続きを記憶し、このオブジェクトに対応してメタオブジェクトが個別にメタデータおよびメタ手続きを記憶している。このため、メタオブジェクト単位でメタデータやメタ手続きの変更が容易にでき、オブジェクトの定義に影響を与えることなく、ユーザの要求や、システムのハードウェア構造、使用する入出力機器等に応じて容易にシステムの動作を変更することが可能である。

【0095】また、オブジェクト毎に異なる動作をするメタオブジェクトを対応させることができるので、同一種類のオブジェクトであっても異なる動作を実現でき、様々な要求に容易に答えられるという効果がある。さらに、オブジェクトを手続きと独立して容易に定義できるので、オブジェクト指向データベースの開発効率が向上するという効果がある。

【0096】本実施例では、メモリ上のオブジェクトのデータの値や構造が変更された場合には、その都度または一定時間の経過毎にディスクに記憶する。従って故障等によりシステムが突然停止しても、最新のオブジェクトのデータがディスク上に保存され、オブジェクトおよびメタオブジェクトを永続的に記憶できるという効果がある。

【0097】一方、オブジェクトの手続きやメタオブジェクトの変更は、必ずしもその都度ディスクに反映することをせず、システム立ち下げ時に反映することで、処理効率を向上している。これらメモリ上のオブジェクトデータのディスク上への記憶処理は、メタオブジェクトがメタデータを参照しながら、メタ手続きにより行っているので、オブジェクトに対応して所望のメタ手続きを設定することで、ユーザの要求やシステムに適応してカスタマイズできる。

【0098】本実施例では、プログラマがオブジェクトを実行時に動的に変更することが可能である。すなわち、メタオブジェクトは、オブジェクトのデータや手続きの記憶領域や、手続きの呼び出し動作に関する情報を有しているので、メタ手続きはこれを参照することにより、データや手続きの追加や削除、変更など、オブジェクトの動的処理をプログラマが自由にカスタマイズできる。

【0099】次に、本発明の他の実施例について説明する。上記した本発明の実施例では、個々のオブジェクトがそれぞれ個別に手続きを有していた。しかし、同じ種類のオブジェクト、すなわち、同じクラスに属するオブ

ジェクトは、同一の手続きを実行すればよいので、手続きを共有できる。

【0100】図14に、共有する手続きのクラスをもつオブジェクトの構造を示す。このクラス<sub>11cm</sub>は、手続き<sub>11p10</sub>、<sub>11p20</sub>等の手続きのプログラムの先頭番地を示す手続きアドレス<sub>11p1</sub>、<sub>11p2</sub>等を記憶する。そして、オブジェクト<sub>110m</sub>は、データ<sub>11d1</sub>、<sub>11d2</sub>等と、そのオブジェクトが属するクラス<sub>11cm</sub>の記憶領域へのポインタ<sub>11cp</sub>を記憶する。同一クラスに属するオブジェクト<sub>120m</sub>も、同一クラス記憶領域へのポインタ<sub>12cp</sub>を記憶する。これにより手続きを記憶するための記憶領域を小さくできる。

【0101】図15は、本実施例によるメタオブジェクトの構造を示したものである。メタオブジェクトにおいても、同様に、メタ手続きを共有することが可能である。また、同一クラスのオブジェクトに対応するメタオブジェクトは、メタデータの一部を共有できる。例えば、オブジェクトのサイズや、データのアドレス<sub>21d814</sub>を除くデータデータ情報<sub>21d81</sub>、手続き情報<sub>21d101</sub>等のメタデータは共有できる。

【0102】図15に示すように、メタオブジェクトのクラス<sub>21cm</sub>を設け、ここに、メタ手続きのプログラム<sub>21p10</sub>、<sub>21p20</sub>等の先頭アドレス<sub>21p1</sub>、<sub>21p2</sub>等を記憶するとともに、共有データ<sub>21dc1</sub>、<sub>21dc2</sub>等を記憶する。メタオブジェクト<sub>210m</sub>は、共有データ以外のメタデータ、すなわち、個別メタデータ<sub>21dd1</sub>、<sub>21dd2</sub>等と、そのメタオブジェクトが属するクラス<sub>21cm</sub>の記憶領域へのポインタ<sub>21cp</sub>を記憶する。同一クラスに属するオブジェクト<sub>220m</sub>も、同一クラス記憶領域へのポインタ<sub>22cp</sub>を記憶する。これによりメタ手続きおよびメタデータを記憶するための記憶領域を小さくできる。

【0103】このメタオブジェクトに対するアクセス方法は、当該メタデータが共有データか個別データかを識別することにより可能になる。図16に、本実施例によるメタデータのアクセス処理<sub>21p500</sub>を示す。まず、アクセスするメタデータが共有かどうか判定し(処理<sub>21p501</sub>)、共有であれば共有メタデータをアクセスし(処理<sub>21p502</sub>)、そうでなければメタオブジェクト毎に記憶されている個別メタデータをアクセスする(処理<sub>21p503</sub>)。

【0104】共有または個別メタデータへのアクセス後は、第1の実施例に説明したと同様に、メタオブジェクト単位でメタデータやメタ手続きの変更が容易にでき、オブジェクトの定義に影響を与えることなく、ユーザの要求や、システムのハードウェア構造、使用する入出力機器等に応じて容易にシステムの動作を変更することが可能である。また、メタ手続きにより、データや手続きの追加や削除、変更など、オブジェクトの操作を動的に

変更できる。

【0105】なお、プログラマが手続きの記述時に共有／個別を明示的に記述するか、コンパイル時に判定することにより、共有メタデータへ直接アクセスする処理を、メタ手続きの実行可能プログラム中に含むこともできる。

【0106】以上に説明した本発明の実施例では、磁気ディスク上のオブジェクトおよびメタオブジェクトの複製を、システム立ち上げ時に全てメモリ上に格納する。しかし、システム立ち上げ時ではなく、アプリケーションプログラムが必要とする時に、必要とするオブジェクトまたはメタオブジェクトの複製のみをメモリ上に格納するようにしてもよい。これにより、必要とするオブジェクトとメタオブジェクトのみがメモリ上に存在することとなり、メモリの使用効率が向上する。

【0107】また、オブジェクトのデータについては、変更の都度ディスクに反映し、オブジェクトの手続きやメタオブジェクトの変更についてはシステム立ち下げ時にディスクに反映していた。しかし、全てのデータを変更の都度ディスクに反映してもよい。これにより、オブジェクトやメタオブジェクトの最終状態を常に保存し、システムが電源ダウンで突然停止し、非常電源により直ちに再立ち上げする場合に、立ち上げ時の再生処理をする必要がなく、システムの動作を瞬時に回復することができる。

【0108】また、メモリ上のメタオブジェクトの情報をシステム立ち下げ時にディスクに書き込んでいた。しかし、オブジェクトのデータ構造や手続きが変更されなければ、メタデータは変更されない。そこで、メタオブジェクトの情報についてはディスクに書き込まないようすることもできる。これにより処理効率を向上できる。

#### 【0109】

【発明の効果】本発明によれば、オブジェクトのデータ構造や手続きを修正することなく、メタオブジェクトを修正するのみで、オブジェクトの動作を変更することができる。また、オブジェクトとは独立に、オブジェクトに関する処理をメタオブジェクトにより追加できる。これにより、ユーザの要求や、システム構成の変更等に容易に対応でき、カスタマイズが容易になるという効果がある。

【0110】本発明によれば、メタ手続きによって、オブジェクトの永続的記憶装置と一時的記憶装置のデータの記憶ないしデータの一一致化処理を、再生不可能なデータはその都度、再生可能なデータはシステム立ち下げ時に実行させ、システムの信頼性と処理効率を向上できる効果がある。また、ユーザが記憶方法を、容易に変更することができ、オブジェクト指向データベースの移植が容易になるという効果がある。

【0111】本発明によれば、オブジェクトの記憶位置

や記憶構造の変更、オブジェクトのデータや手続きの追加、削除、変更等を、実行時にシステムを停止することなく動的に行うことができる。これにより、システムの可用性が向上するという効果がある。

### 【図面の簡単な説明】

【図1】本発明の一実施例によるオブジェクト指向データベースシステムの構成図。

〔図2〕一実施例のオブジェクトの構造を示す説明図

【図3】一実施例のメタオブジェクトの構造を示す説明図。

【図4】一実施例のメタデータの構造を示す説明図

【図5】一実施例のメタデータの手続き情報の構造を示す説明図

【図6】本発明の一実施例の処理方法で、データ書き込み前メタ手続きの処理を示す流れ図

【図7】データ書き込み前メタ手続きの他の処理例を示す流れ図。

【図8】本発明の一実施例の処理方法で、データ書き込み後ノク手持きの処理を二通り

【図9】本発明の一実施例の処理方法で、入出力後メタ手書きの初期化処理フロー

【図10】入出力後メタ手続きの他の処理例を示す流れ

【図11】本発明の一実施例の処理方法で、データ追加

【図12】データ追加メタ手続きの他の処理例を示す流れ図。

【図13】本発明の一実施例の処理方法で、手書き変更

【図14】本発明の他の実施例を適用するオブジェクト

#### 11. 決定実現を適用するオンラインショット本30

### \* の構造図。

【図15】本発明の他の実施例によるメタオブジェクトの構造図。

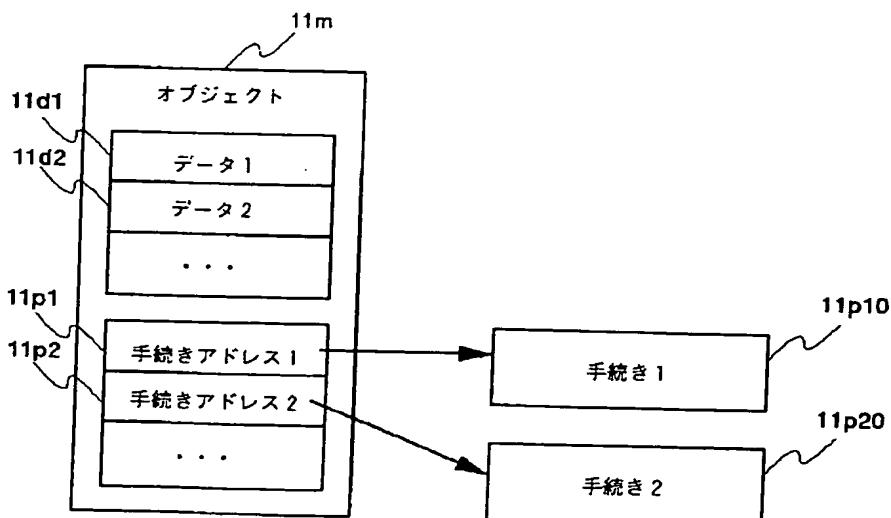
### 【図16】本発明の他の実施例によるメタオブジェクトのアクセス方法を示す流れ図

### 【符号の説明】

- 1, 2…計算機、10…ネットワーク、101…プロセッサ、102…メモリ、103…磁気ディスク装置、104…CRTディスプレイ、105…キーボード、106…ネットワークアダプタ、121…アプリケーションプログラム、122…データベースサーバプログラム、123…オブジェクトキャッシュ領域、124…メタオブジェクト変更プログラム、11d、12d…磁気ディスク装置上のオブジェクト、21d、22d…磁気ディスク装置上のメタオブジェクト、11m、12m…オブジェクトキャッシュ領域上のオブジェクト、21m、22m…オブジェクトキャッシュ領域上のメタオブジェクト、11d1、11d2…オブジェクトのデータ、11p1、11p2…オブジェクトの手続きのアドレス記憶領域、11p10、11p20…オブジェクトの手続きのプログラム、21d1、21d2…メタオブジェクトのメタデータ、21p1、21p2…メタオブジェクトのメタ手続きのアドレス記憶領域、21p10、21p20…メタオブジェクトのメタ手続きのプログラム、11cm…オブジェクトの共有手続きのクラス、11cp、12cp…オブジェクトデータのクラスポインタ、21cm…メタオブジェクトの共有メタデータのクラス、21om、22om…個別のメタデータ、21cp、22cp…共有メタデータのクラスポインタ。

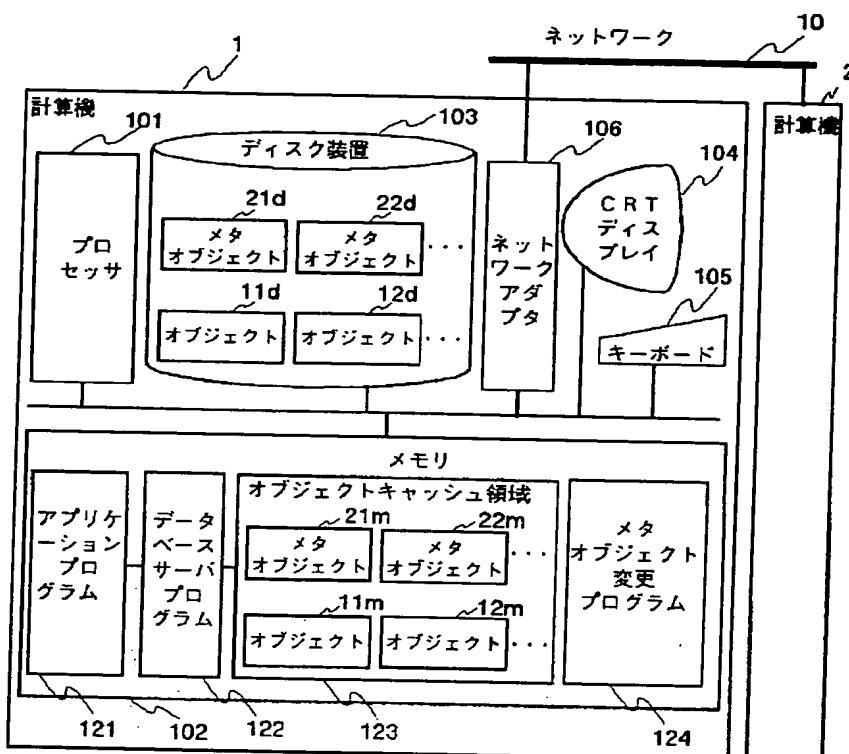
[図2]

2



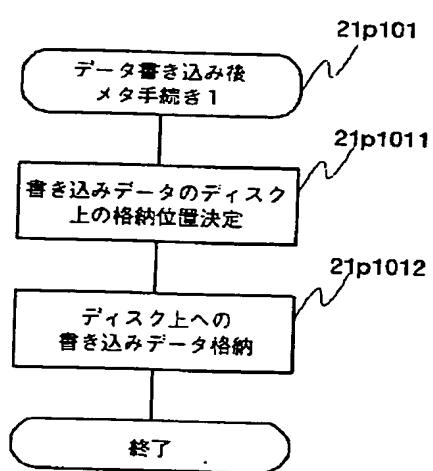
【図1】

図 1



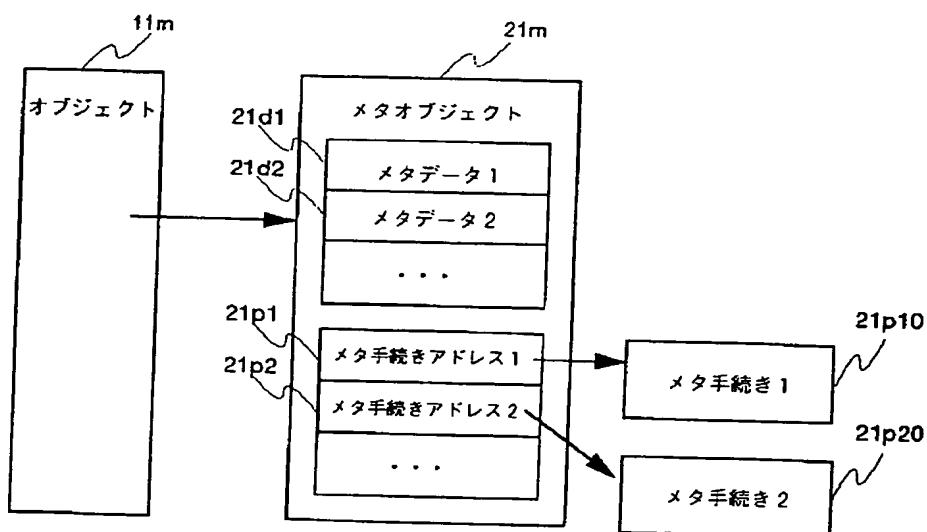
【図7】

図 7



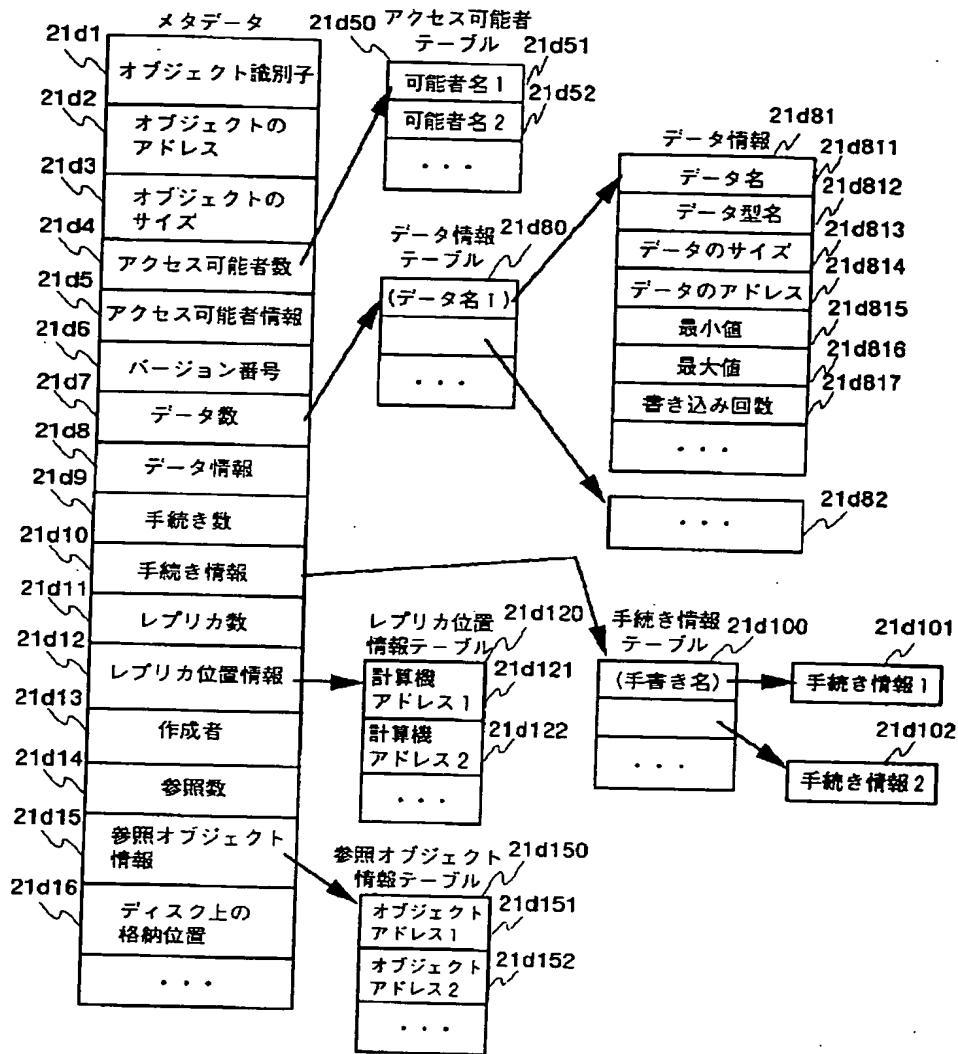
【図3】

図 3



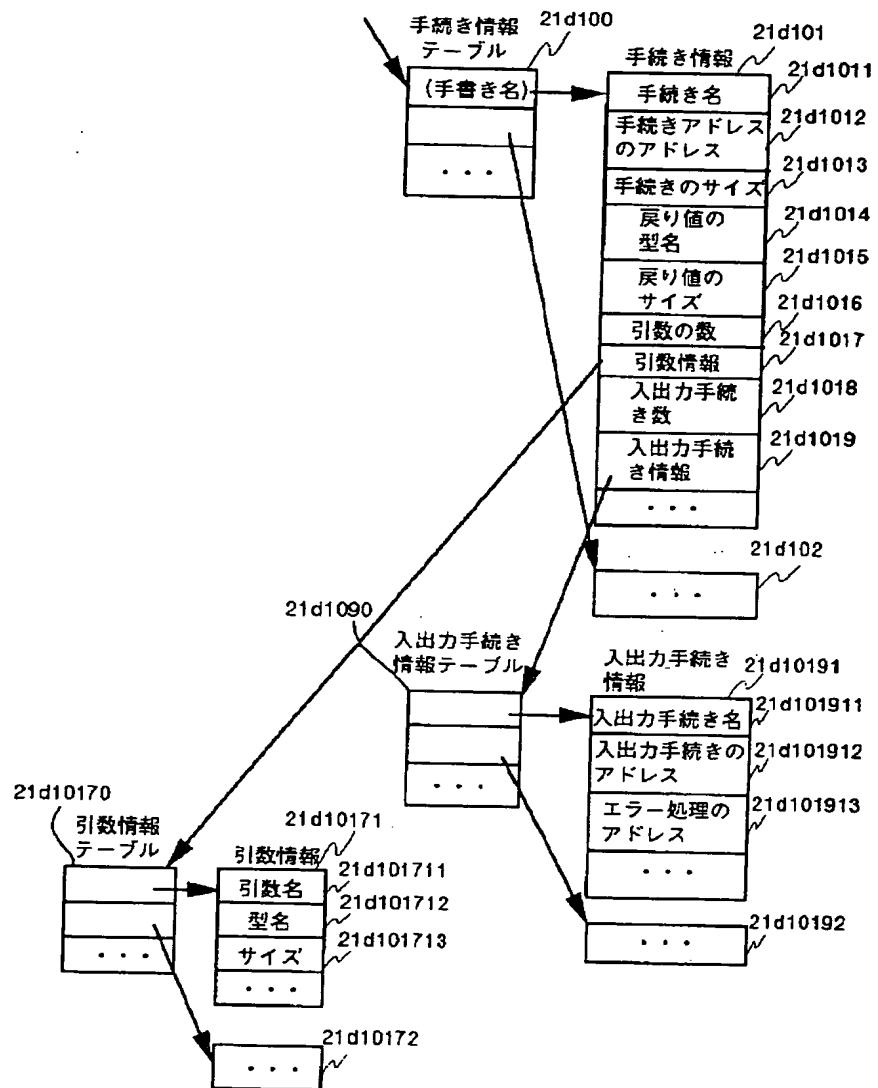
【図4】

図 4



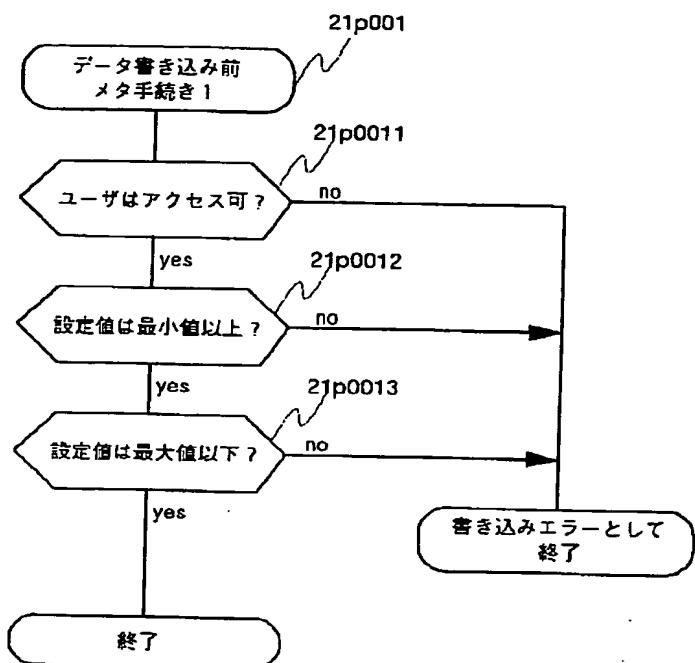
【図5】

図 5



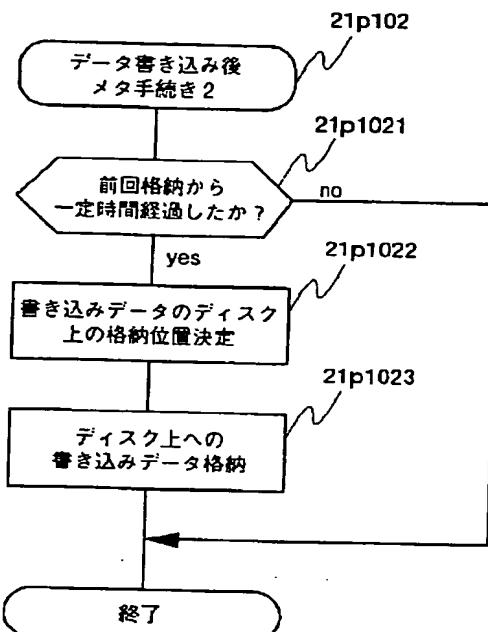
【図6】

図 6



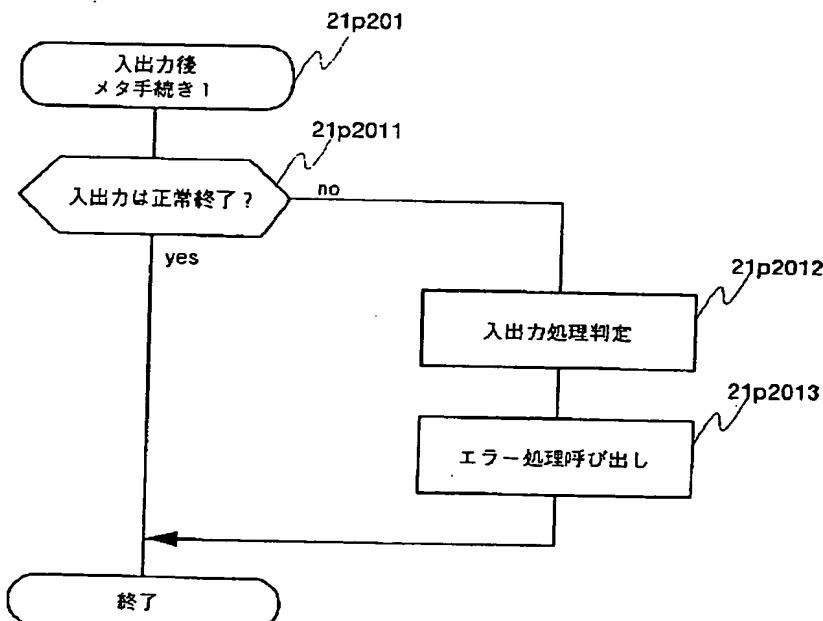
【図8】

図 8



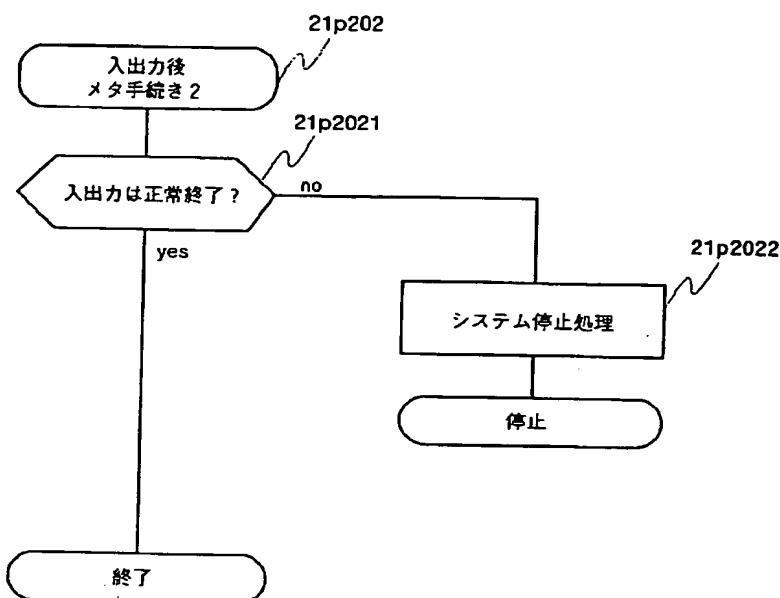
【図9】

図 9



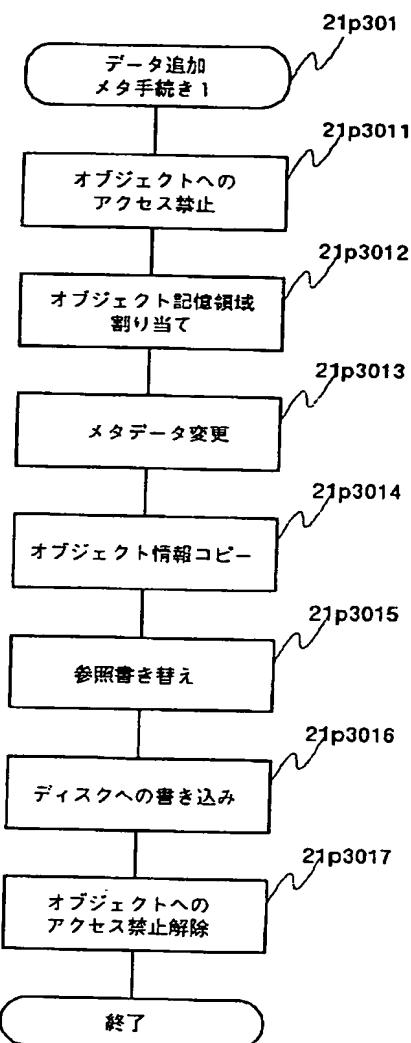
【図10】

図 10



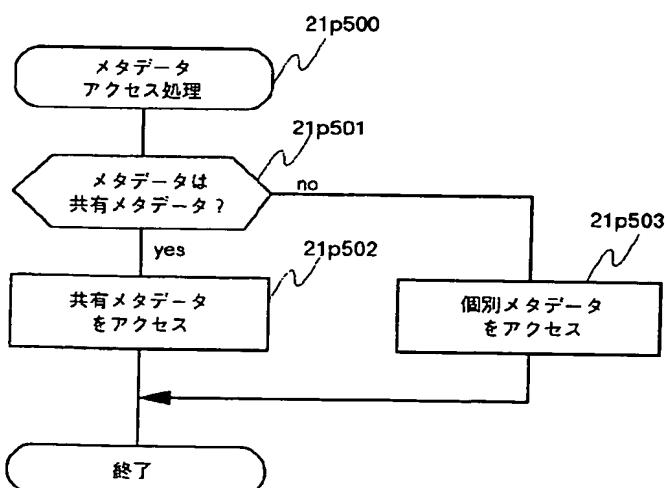
【図11】

図 11



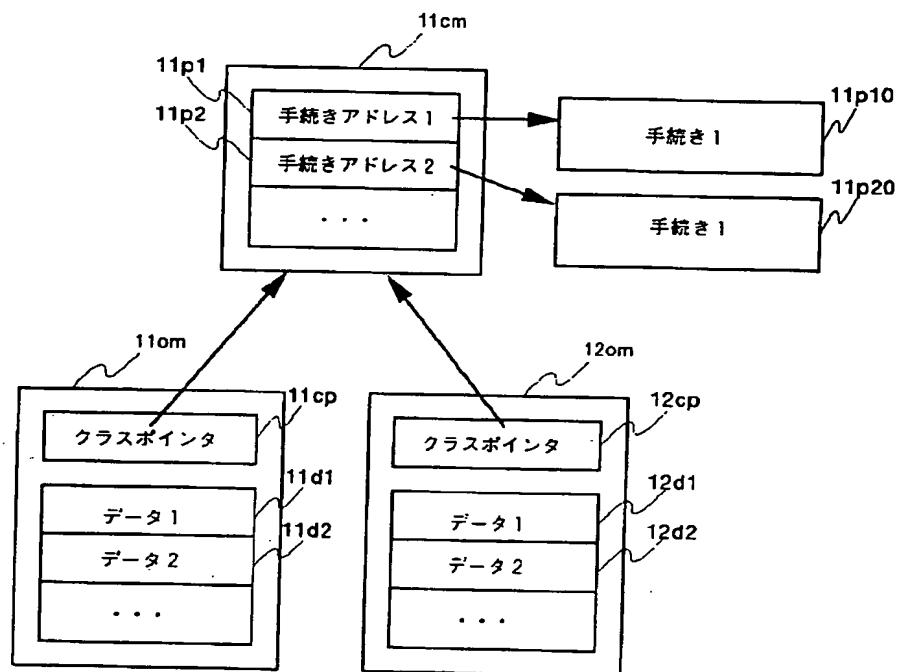
【図16】

図 16



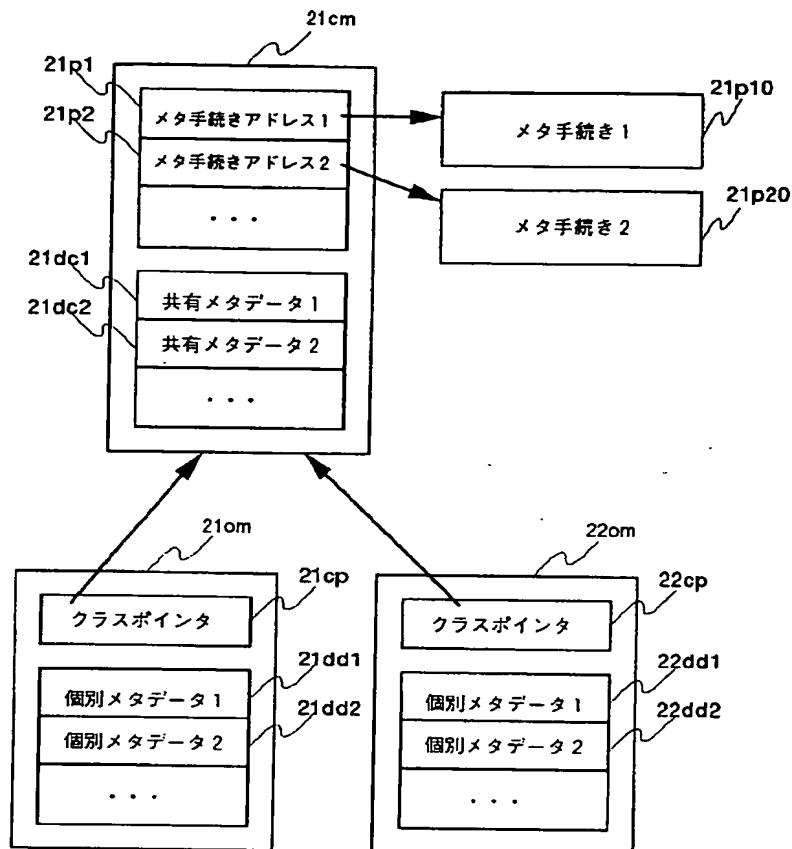
【図14】

図 14



【図15】

図 15



フロントページの続き

(72)発明者 廣田 敦彦  
茨城県日立市大みか町五丁目2番1号 株  
式会社日立製作所大みか工場内

(72)発明者 恒富 邦彦  
茨城県日立市大みか町七丁目1番1号 株  
式会社日立製作所日立研究所内  
(72)発明者 上脇 正  
茨城県日立市大みか町七丁目1番1号 株  
式会社日立製作所日立研究所内

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record.**

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- BLACK BORDERS**
- IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- FADED TEXT OR DRAWING**
- BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- SKEWED/SLANTED IMAGES**
- COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- GRAY SCALE DOCUMENTS**
- LINES OR MARKS ON ORIGINAL DOCUMENT**
- REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- OTHER:** \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**